

TP : Listes Chaînées

POOA – Master TSD1

Exercice1

Partie A

Soit la définition de la structure de donnée suivante destinée à être un nœud d'une liste chaînée.

```
public class Personne{  
  
    public String nom ;  
    public String CIN ;  
    public int age ;  
    public Personne suivant ;  
  
    public Personne(String a, String b, int x, Personne p){  
  
    }  
}
```

soit la déclaration de la liste chaînée suivante :

```
public class ListePersonne{  
  
    Personne premier ;  
  
    .....  
}
```

I- Terminer la déclaration de les classes ListePersonne en ajoutant les méthodes suivantes

- 0- public int longueur() ;
- 1- public void afficheListe() ;
- 2- public boolean recherchePersonne(String cin) ;
- 3- public Personne recherchePersonne(String cin) ;
- 4- public boolean ajoutPersonne(Personne p) ;
- 5- public boolean ajoutPersonne() ;
- 6- public boolean supprimerPersonne(Personne p) ;
- 7- public boolean supprimerPersonne() ;

II- Écrire une classe TestListePersonne qui comporte un main pour tester les différentes méthodes de la classe ListePersonne.

Partie B

Ajouter à la classe ListePersonne deux méthodes :

- 1- une méthode qui permet de trier la liste chaînée selon un ordre croissant par rapport à l'attribut nom.

2- une méthode qui permet d'afficher la liste chaînée triée selon un ordre croissant par rapport à l'attribut nom, sans la modifier.

Modifier le main de la classe TestListePersonne afin de tester les méthodes de trie que vous avez ajouté.

Exercice 2

Nous voulons organiser une bibliothèque musicale sous forme d'une liste chaînée un peu particulière

une première structure désigne un album comportant deux chaînes de caractères désignant son titre et l'artiste ou le groupe interprète. En plus d'un attribut qui désigne l'album suivant.

Une deuxième structure désigne un genre musical comportant une chaîne de caractère genre et le genre musical suivant et l'adresse du premier album.

```
public class Album{

    String titre ;
    String artiste ;
    Album suiv ;

    .....

}

public class GenreMusical{

    String genre ;
    GenreMusical suiv ;
    Album premier ;

    .....

}
```

1- Écrire une structure Mediatheque qui comporte un seul attribut GenreMusical premier et ajouter les fonctions d'affichage, d'ajout, de suppression, de recherche et de modification pour

- un album
- un genre musical

ajouter une fonction qui permet d'afficher toute la Médiathèque.

2- Dans une classe TestMediatheque, écrire une fonction main qui permet de tester les différentes opérations possibles sur la médiathèque.